

USCMS T2 Meeting

Grid Security Issues

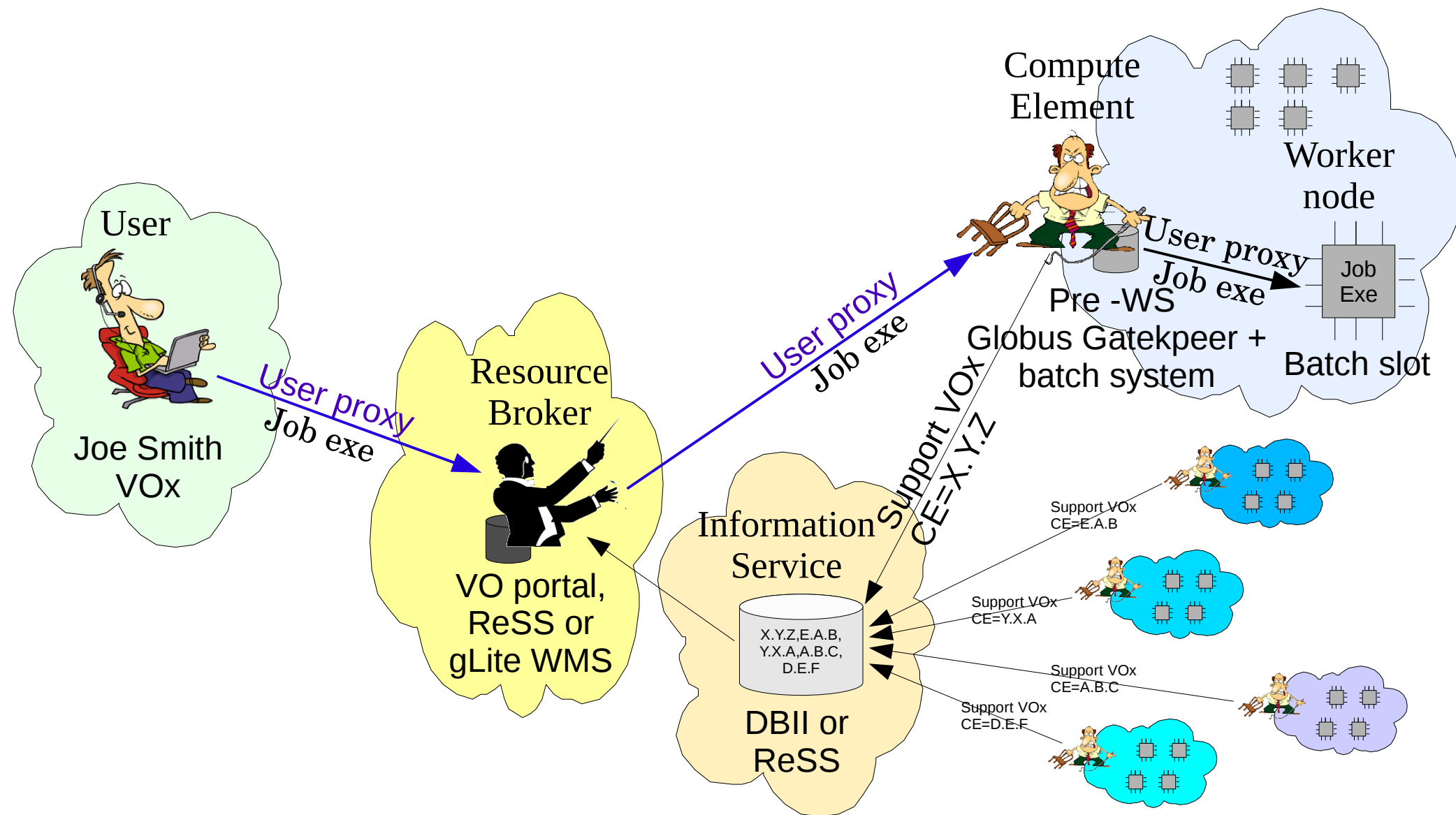
**Why the Grid security model is broken
and
How can we fix it**

by Igor Sfiligoi^{Fermilab}

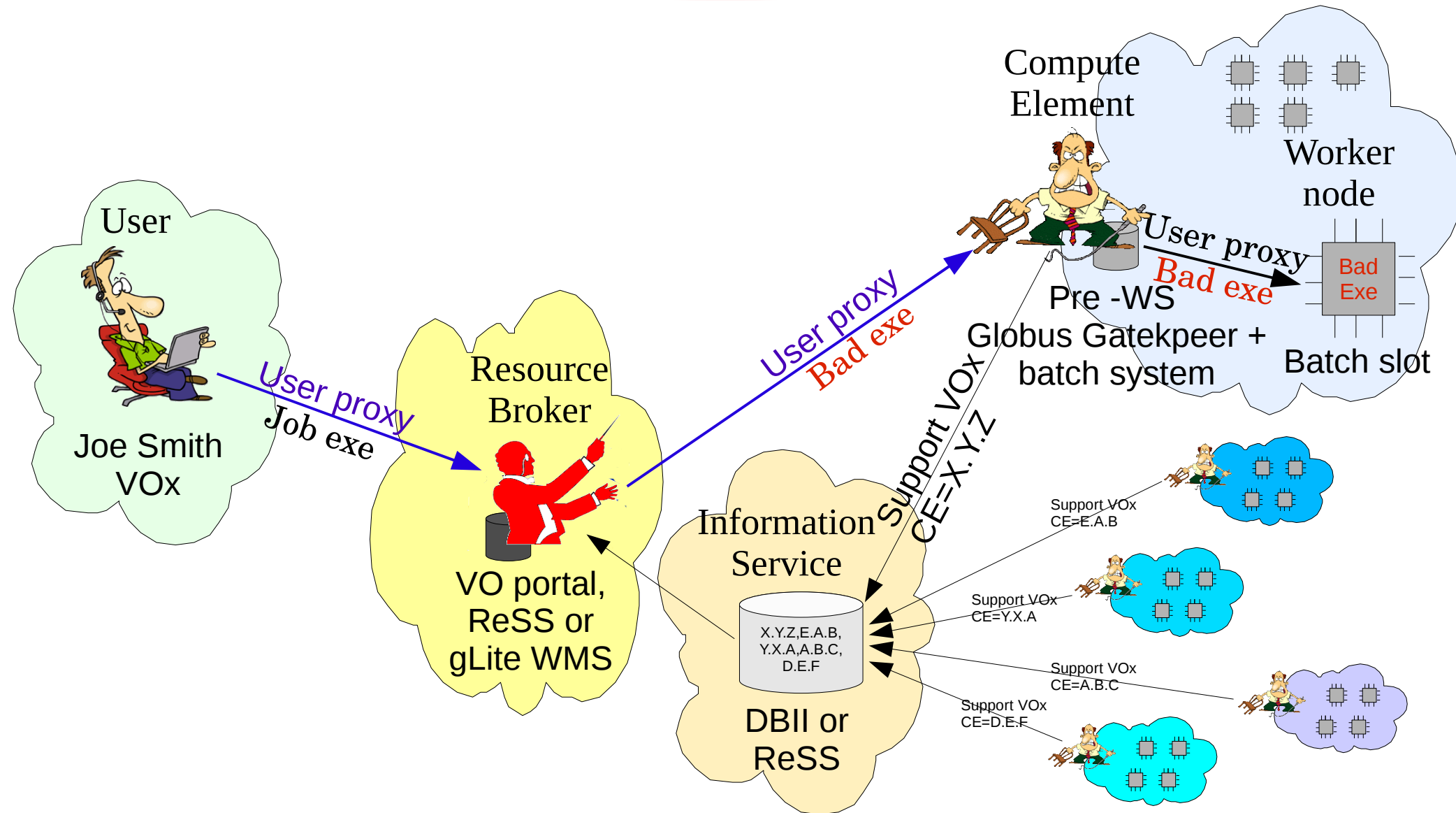
Outline

- Typical job workflows
- Security problems
- Proposed solutions
- Conclusions

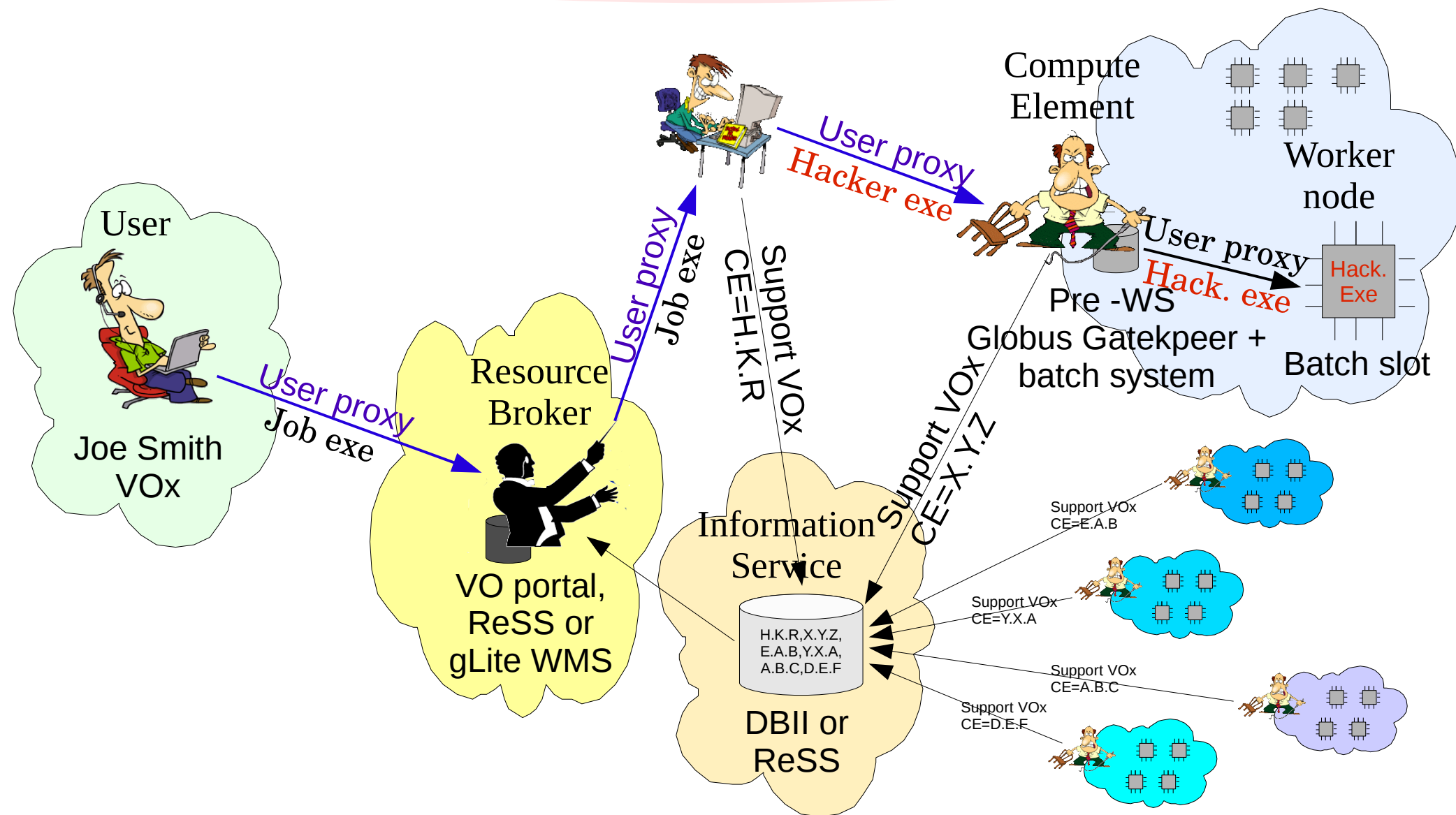
A classical job workflow



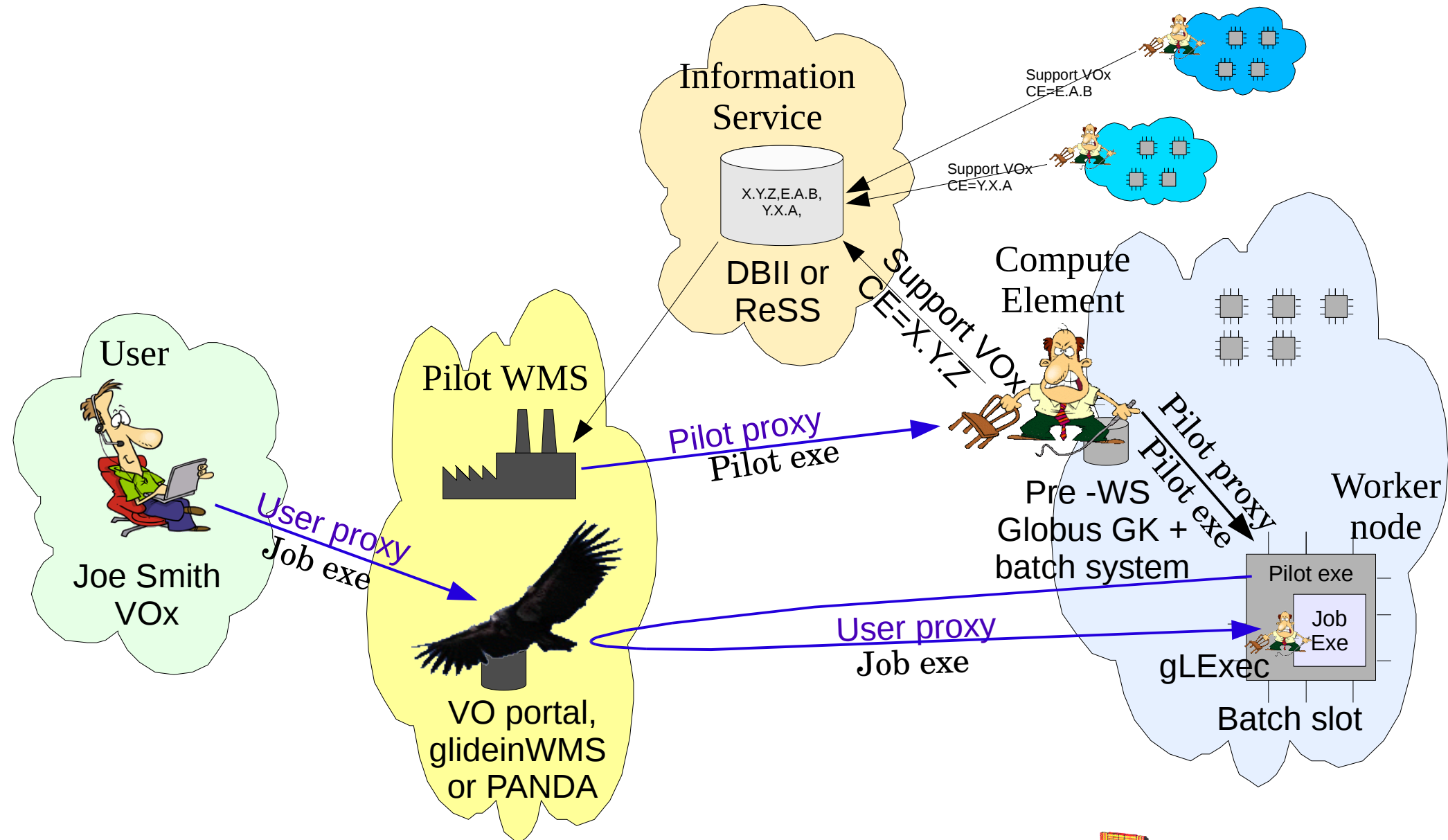
A compromised RB



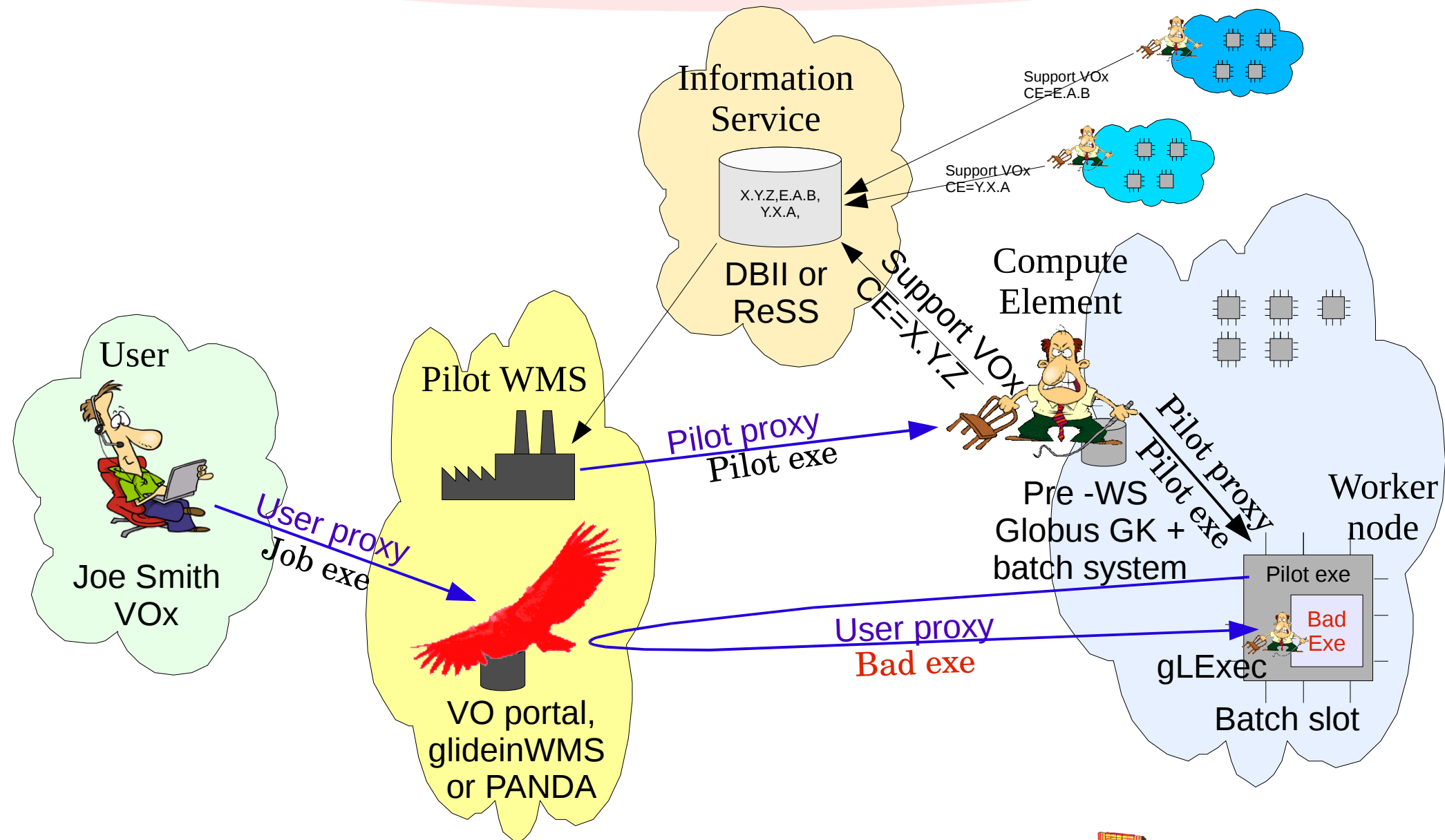
A hacker poses as a site



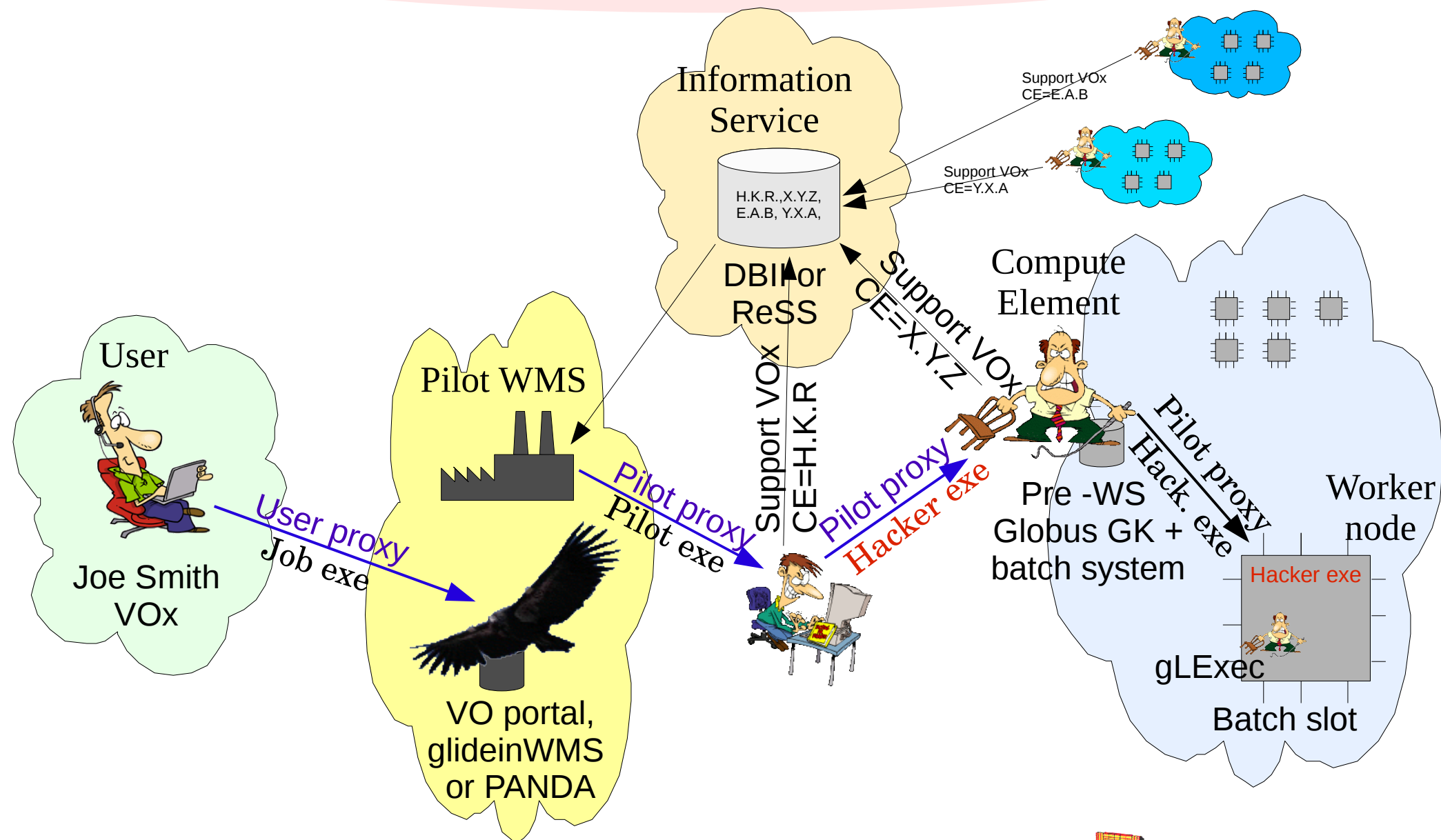
A pilot job workflow



A compromised pilot WMS

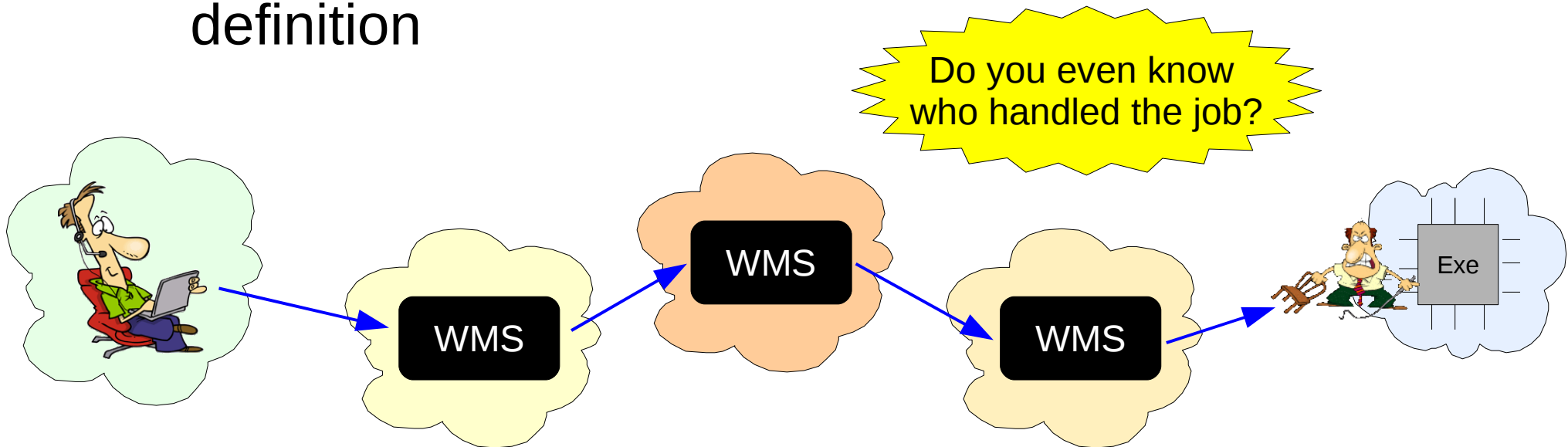


A hacker poses as a site



Mid-talk summary

- All entities in the Grid **MUST** be trusted
 - User delegates all rights to the intermediaries
- Establishing trust can be problematic
 - Grids span multiple administrative domains by definition

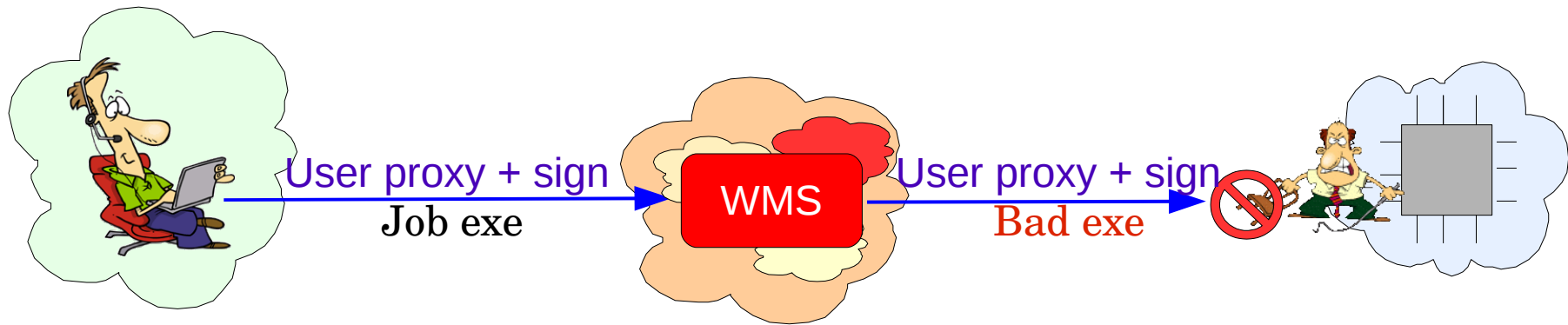


Current protections

- The only protection is the lifetime
 - Proxies are supposed to be short (order of an hour)
 - Limits the time in which the hacker can do damage
- Real life proxies are not short
 - Definitely not a single hour, often weeks
 - Good reasons for this
 - Users submit jobs that will finish after weeks

My proposal

- Embed the executable signature into the proxy
 - Something like MD5SUM or SHA1SUM
 - Proxy attributes cannot be modified
- Check signature at gatekeeper (or gLExec)
 - Refuse to run a job that does not match signature



How would it work?

- Need changes to end-points
 - For example, Condor-G submit and gatekeeper
- **condor_submit** embeds the signature before sending the job and proxy to **condor_schedd**
 - User needs to thrust the node and the local condor_submit only for the duration of the submission
- The gatekeeper (gLExec on WN) checks the signature before executing the job
 - Owned by root => node owner trusts it

Benefits

- No need to trust the intermediaries
 - Site can check the job signature
- Incrementally deployable
 - Middleware can ignore the signature
 - Proxies with signature can be trusted more
 - For example, allowing longer proxy lifetimes
- May allow to get rid of short proxy lifetime limits
 - A stolen proxy w/signature is not a big threat

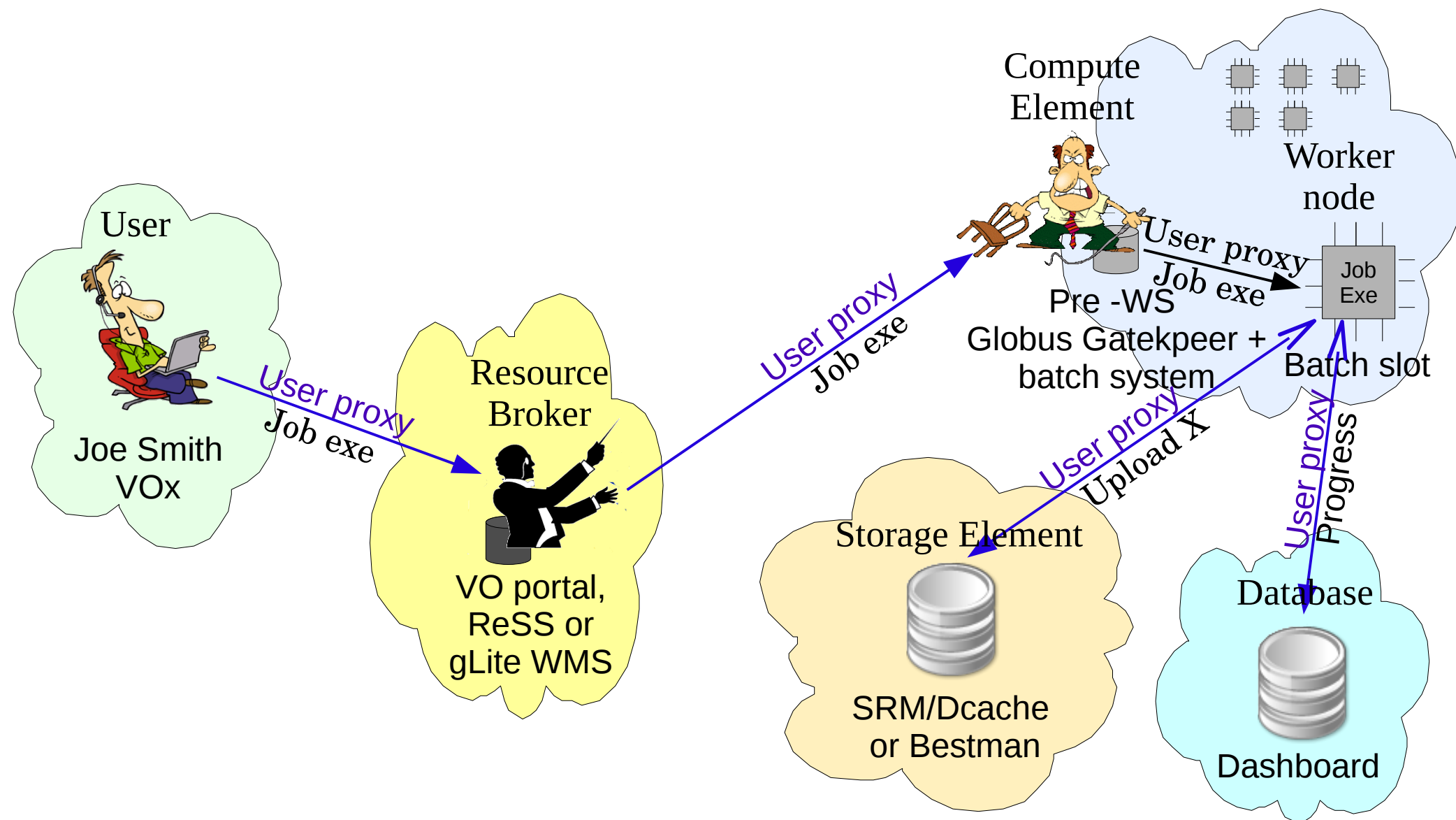
How to get there?

- Define the content to be signed
 - Executable, arguments, input files, other?
- Define a representation for the signature
 - Single signature?
 - One signature per element? (what language?)
- Implement it in commonly used tools
 - Condor and gLite WMS clients
 - Globus Gatekeeper and gLExec

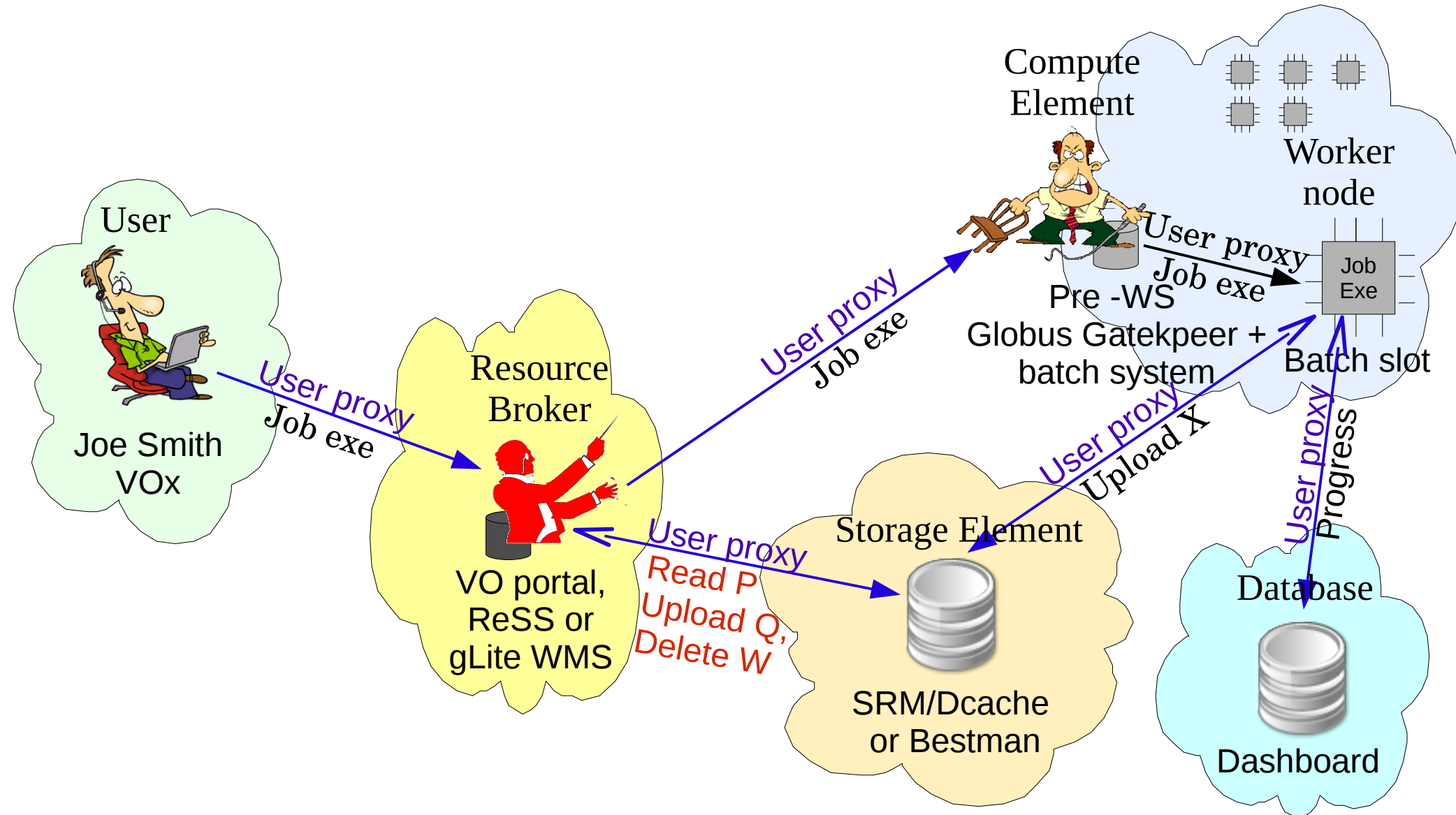
Part Two

- Preventing arbitrary code execution is just part of the problem
 - **The most important one in my opinion**
 - This is why it was presented first
- A proxy with the exe signature still has to be used for access to **storage elements** and **databases**

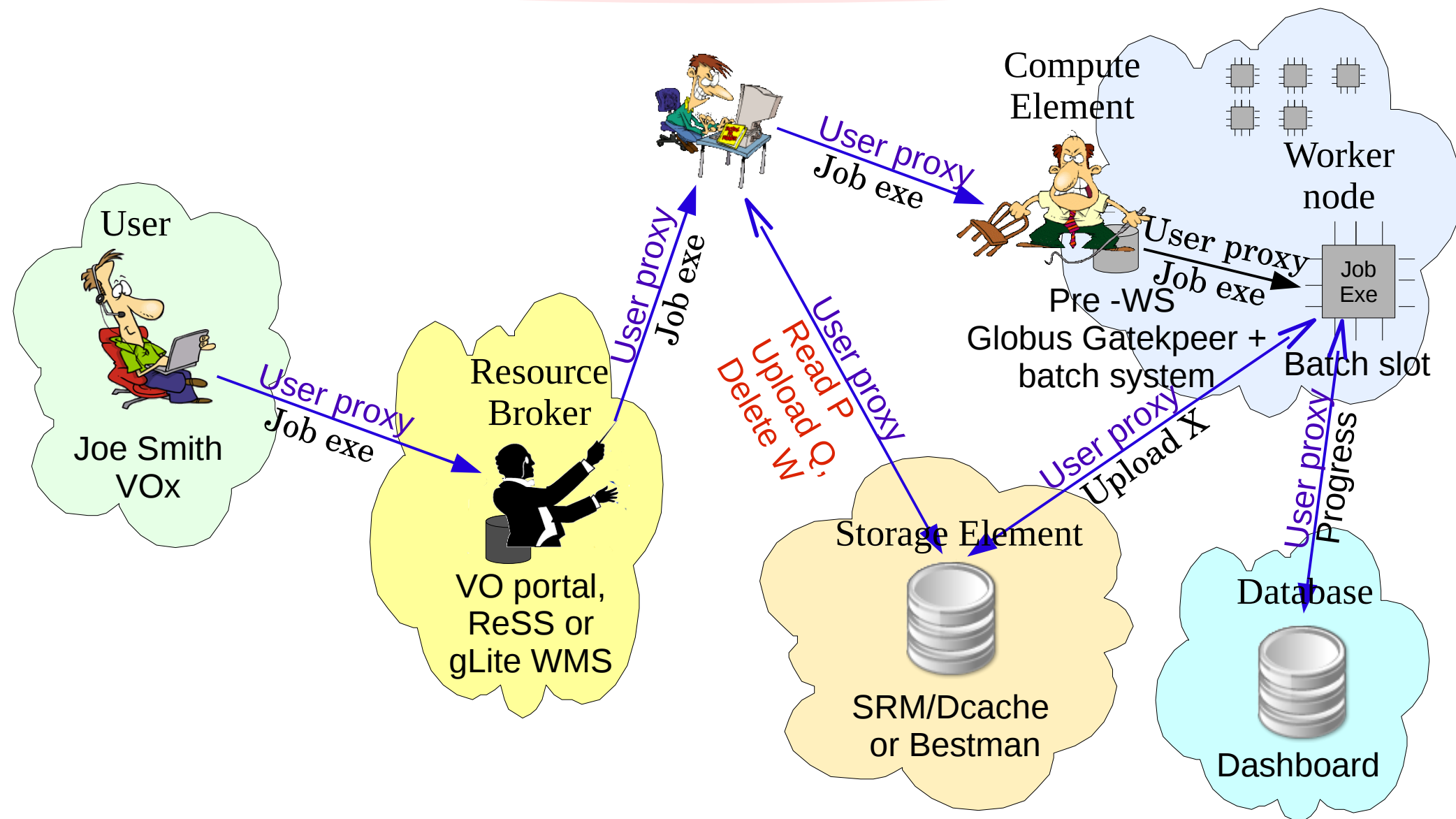
Job accessing data



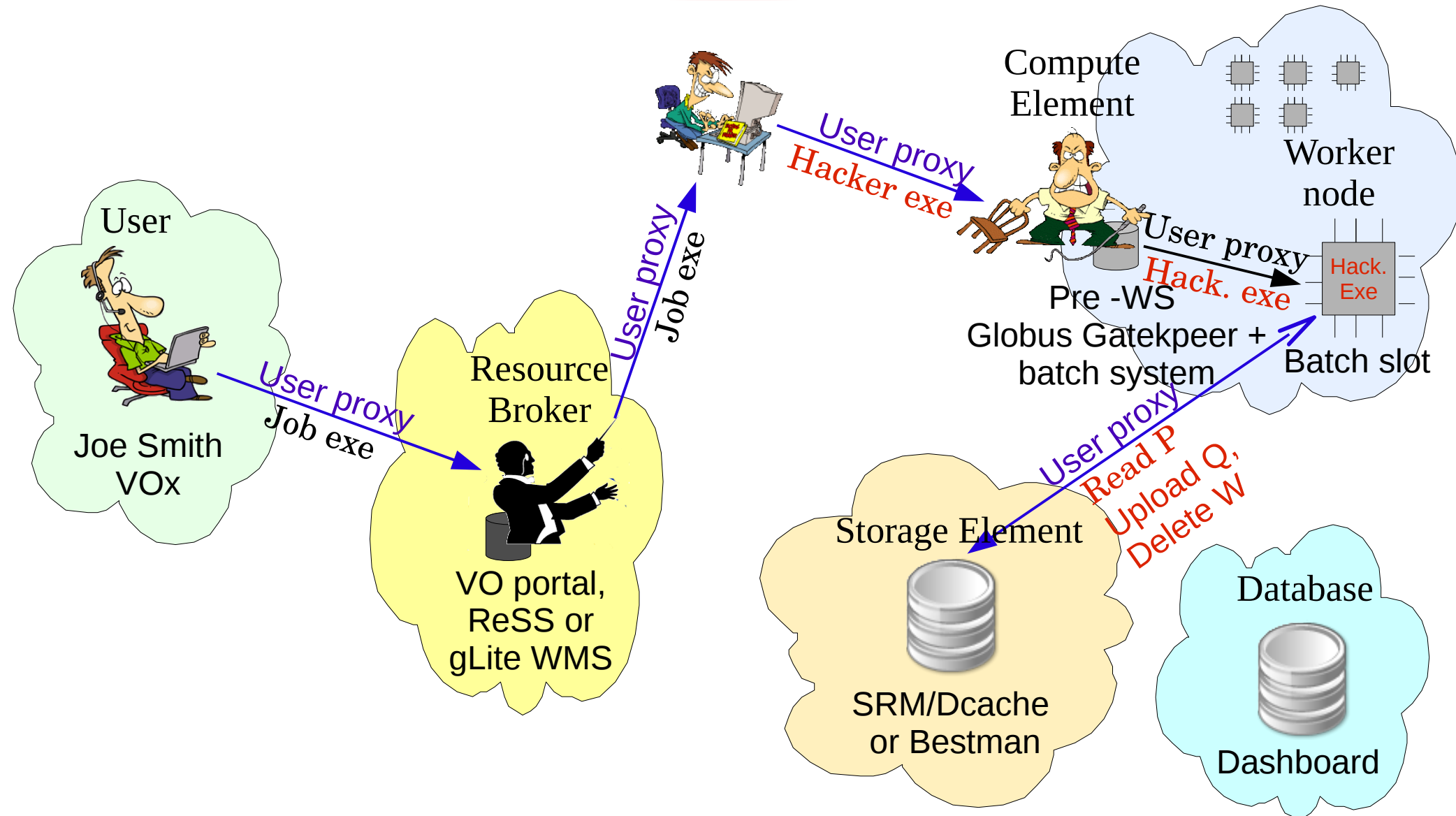
Compromised RB accessing data



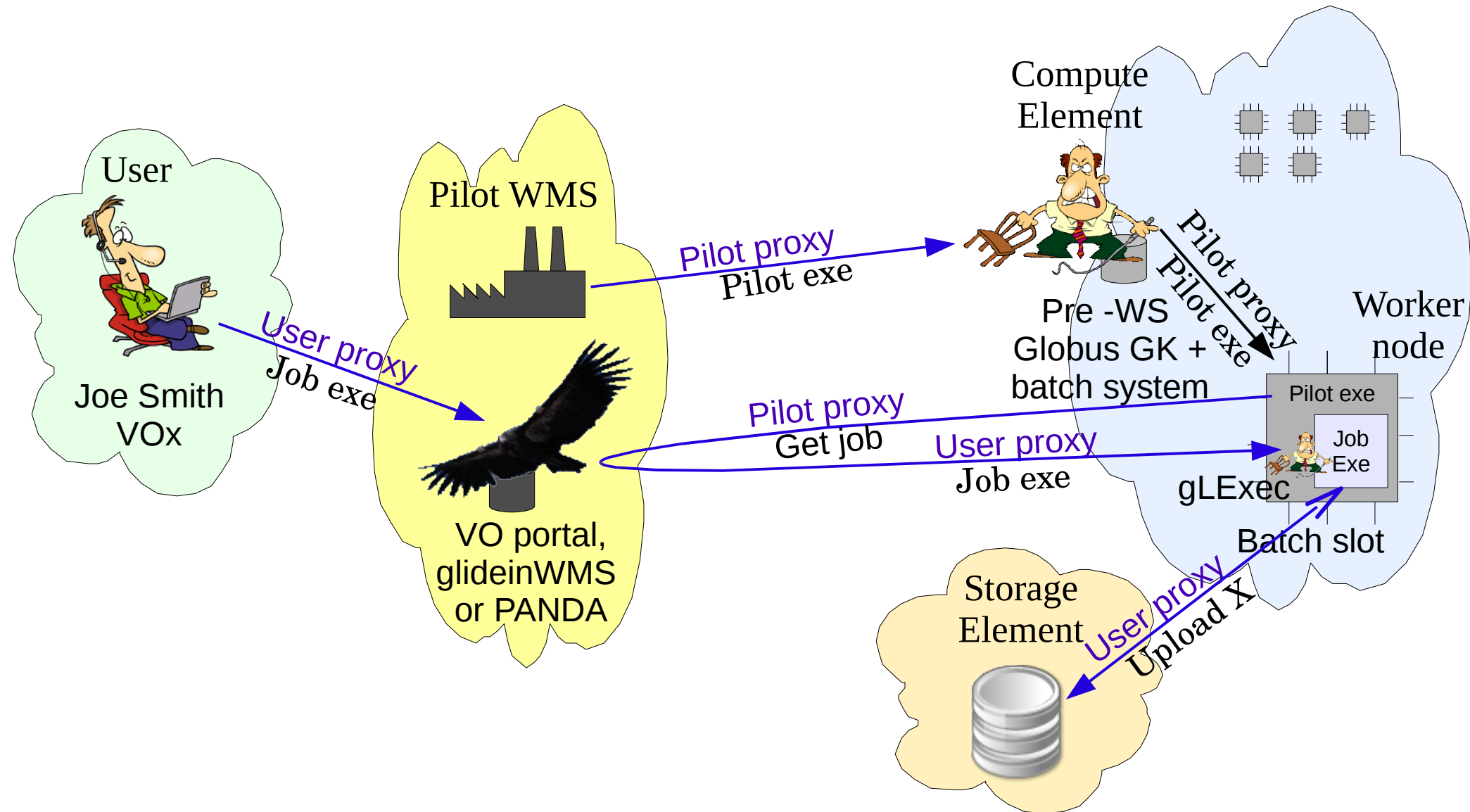
Hacker accessing data



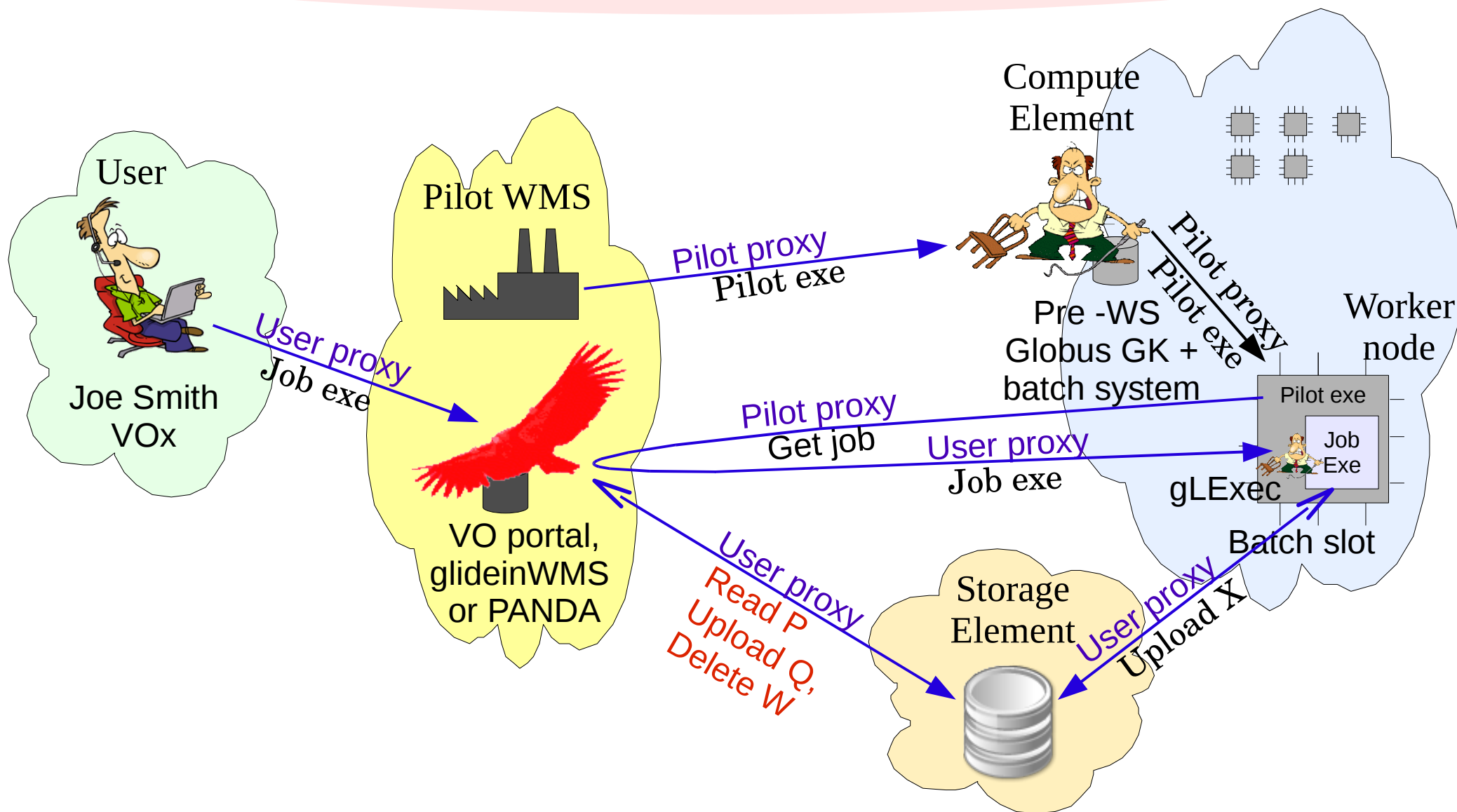
Hacker acc. Data – No signatures



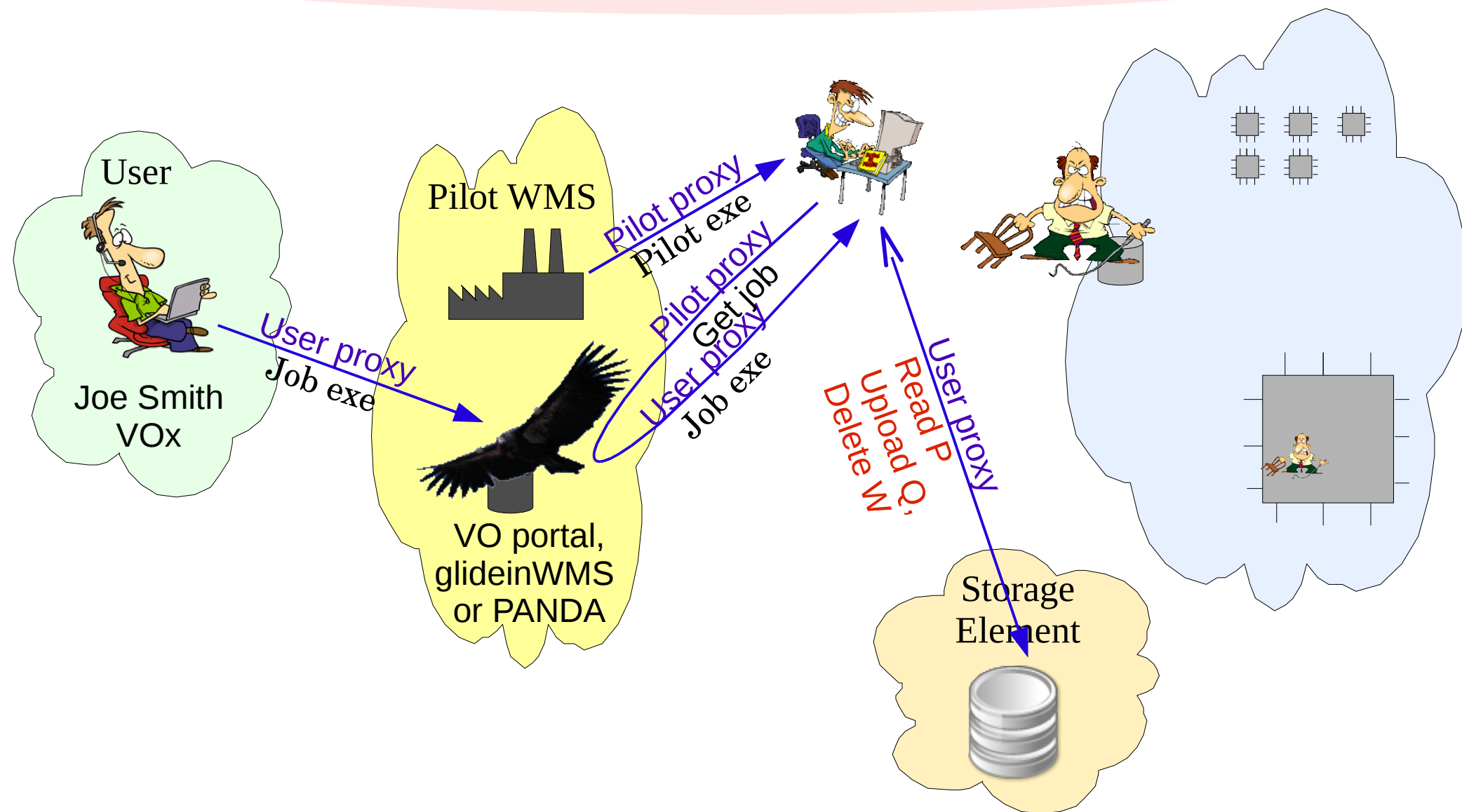
A pilot job accessing Data



A comprom. pilot WMS acc. Data



A hacker in the loop

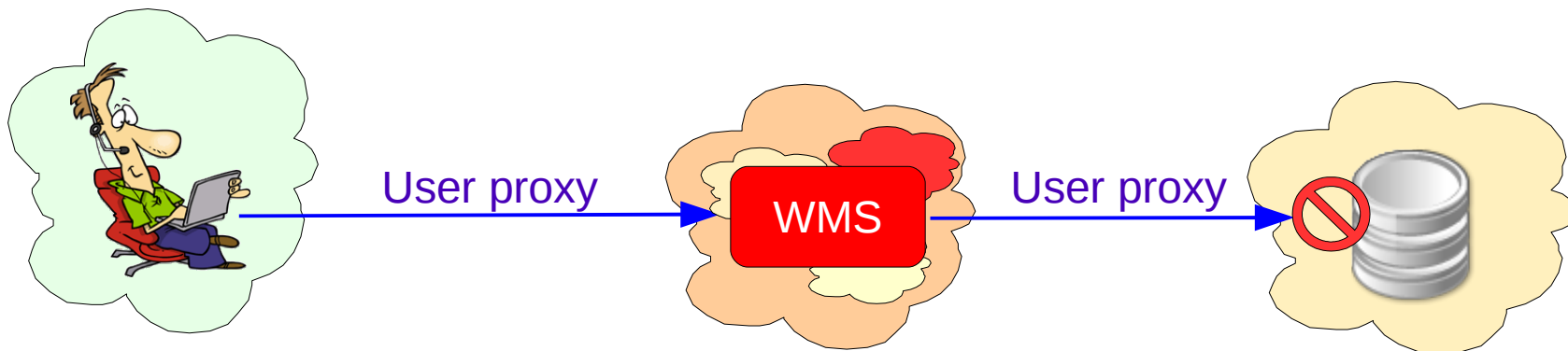


Current protections

- Again, the only protection is the lifetime
 - To limits the time in which the hacker can do damage
- Real life proxies are not short
 - Must last at least the as much as the job

My proposal

- Restrict the access actions of the delegated proxy
 - For example, a MC job does not need to delete files!
- Allow only connections from trusted resources
 - User must trust them to run his/her jobs anyhow



How do we do it?

- **Good question!**
- Limiting actions is feasible (user specifies)
 - Assuming users would bother using it
 - Should be automated (R&D)
- Keeping a list of trusted resources:
 - Far from obvious at this time
 - Need R&D to get a reasonable solution

Conclusions

- Current Grid security mechanisms require a lot of trust to work well
- Using an exe signature would greatly limit damage from compromised proxies
 - Relatively easy to implement
- More needs to be done to protect the data, too
 - Needs substantial R&D to get to a concrete solution
- As a side effect, we may allow longer lived proxies

Appendix

- I would like to work on solving this security problems
- I am paid by USCMS
- Need your support to get the project approved
 - Until now spent only a tiny fraction of my time on this